

Hierarchical Color Quantization with a Neural Gas Model based on Bregman Divergences

Esteban J. Palomo^{1,2}, Jesús Benito-Picazo^{1,2}, Enrique Domínguez^{1,2}, Ezequiel López-Rubio^{1,2}, and Francisco Ortega-Zamorano^{1,2}

¹ University of Málaga, Bulevar Louis Pasteur, 35, 29071 Málaga, Spain
{ejpalomo,jpicazo,enriqued,ezeqlr,fortega}@lcc.uma.es

² Biomedical Research Institute of Málaga (IBIMA), Málaga, Spain

Abstract. In this paper, a new color quantization method based on a self-organized artificial neural network called the Growing Hierarchical Bregman Neural Gas (GHBNG) is proposed. This neural network is based on Bregman divergences, from which the squared Euclidean distance is a particular case. Thus, the best suitable Bregman divergence for color quantization can be selected according to the input data. Moreover, the GHBNG yields a tree-structured model that represents the input data so that a hierarchical color quantization can be obtained, where each layer of the hierarchy contains a different color quantization of the original image. Experimental results confirm the color quantization capabilities of this approach.

Keywords: color quantization, clustering, neural networks, self-organization

1 Introduction

One of the most common image processing tasks is color quantization (CQ), which consists of selecting a small but representative set of indexed colors (codebook) for coding the original digital image with minimum perceptual distortion. This process is widely used to reduce the storage requirements and the transmission bandwidth of color images while maintaining an acceptable image fidelity. Furthermore, CQ is also considered as a useful lossy compression method, where the codebook (an acceptable set of colors) can be used to represent the original colors of a digital image. The quality of the codebook depends on the error between the original image and the resultant image. An optimal codebook should minimize this error, which is usually measured by the mean squared error criterion.

In the literature, many well-known algorithms have been proposed to find optimal codebooks such as the k-means algorithm [10], fuzzy c-means [2], competitive learning [6], self-organizing map [8], and their variants. The Self-Organizing Map (SOM) [7] was the starting point for the development of many self-organizing models [11, 12, 14]. Some of them try to face some drawbacks of the original SOM regarding its pre-established network architecture, i.e. topology and number of neurons [9, 13, 17]. The Growing Hierarchical Self-Organizing Map (GHSOM) [18]

represents a hierarchical extension of the SOM to reflect hierarchical data, where the entire architecture of the neural network is automatically determined during the unsupervised learning process. The Growing Neural Gas (GNG) [5] constitutes another successful variant of SOMs that solves the fixed-network architecture problem by providing a neuron growth and removal mechanism to automatically determine the number of neurons during the unsupervised learning process according to the input data. The GNG has also a hierarchical extension called the Growing Hierarchical Neural Gas (GHNG) [15], in which a neuron can be expanded into a new graph in a subsequent layer of the hierarchy depending on the quantization error associated to that neuron or the graph it belongs to.

These hierarchical models can reflect hierarchical relations present among input data in a more straightforward way. Moreover, the hierarchical architecture provides different quantization rates, depending on the number of levels used. In many image processing applications, the user can select the compression rate or size when an image is saved in a compressed format (e.g. jpeg). This selection is related to the size of the codebook to be used when the image is compressed. Using these kinds of hierarchical models allows to evaluate the quality of the resultant image with different compression rates and to select the best choice.

The use of the Euclidean distance is prevalent in most of the proposed algorithms for CQ. However, this distance may not be the most suitable for all input distributions. In this sense, Bregman divergences are considered in this paper for CQ because their minimizer is the mean [1] and the squared Euclidean distance is a particular case of the Bregman divergences. Therefore, the most suitable divergence can be specified according to the input data by using Bregman divergences. In this paper, a self-organizing neural network called the Growing Hierarchical Bregman Neural Gas (GHBNG) [16] is used for CQ, which is a variation of the GHNG model which considers Bregman divergences as an extension of the classic Euclidean distance.

The remainder of this paper is organized as follows. Section 2 presents the GHBNG model. After that, Section 3 describes the color quantization application and reports the obtained experimental results. Finally, some important conclusions of this work are extracted in Section 4.

2 The GHBNG Model

This work is based on the Growing Hierarchical Bregman Neural Gas (GHBNG) network [16], which can be regarded as a Growing Hierarchical Neural Gas (GHNG) model [15] where Bregman divergences are employed to calculate the winning unit. Let $\phi : \mathcal{S} \rightarrow \mathbb{R}$ be a function with real values and strictly convex, which is defined on a convex set $\mathcal{S} \subseteq \mathbb{R}^D$, where D is the input data dimension [3, 4, 19]. It is further assumed that ϕ is differentiable within the relative interior $\text{ri}(\mathcal{S})$ of \mathcal{S} [1]. Under these conditions, the Bregman divergence $D_\phi : \mathcal{S} \times \text{ri}(\mathcal{S}) \rightarrow [0, +\infty)$ associated to ϕ is given by:

$$D_\phi(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x}) - \phi(\mathbf{y}) - (\mathbf{x} - \mathbf{y})^T \nabla \phi(\mathbf{y}) \quad (1)$$

where $\mathbf{x} \in \mathcal{S}$ and $\nabla \phi(\mathbf{y})$ denotes the gradient vector of ϕ for $\mathbf{y} \in \text{ri}(\mathcal{S})$. Table 1 shows the Bregman divergences that are employed in this work.

Table 1. Bregman divergences employed in this work. \mathbb{R}_+^D denotes the set of D sized vectors with strictly positive real components.

Divergence	\mathcal{S}	$\phi(\mathbf{x})$	$D_\phi(\mathbf{x}, \mathbf{y})$
Squared Euclidean distance	\mathbb{R}^D	$\ \mathbf{x}\ ^2$	$\ \mathbf{x} - \mathbf{y}\ ^2$
Generalized I-divergence	\mathbb{R}_+^D	$\sum_{k=1}^D x_k \log x_k$	$\sum_{k=1}^D \left(-x_k + y_k + x_k \log \frac{x_k}{y_k} \right)$
Itakura-Saito distance	\mathbb{R}_+^D	$-\sum_{k=1}^D \log x_k$	$\sum_{k=1}^D \left(-1 + \frac{x_k}{y_k} - \log \frac{x_k}{y_k} \right)$
Exponential loss	\mathbb{R}^D	$\sum_{k=1}^D \exp x_k$	$\sum_{k=1}^D (\exp x_k - \exp y_k - (x_k - y_k) \exp y_k)$
Logistic loss	$(0, 1)^D$	$\sum_{k=1}^D (x_k \log x_k + (1 - x_k) \log (1 - x_k))$	$\sum_{k=1}^D \left(x_k \log \frac{x_k}{y_k} + (1 - x_k) \log \frac{1 - x_k}{1 - y_k} \right)$

Bregman divergences are amenable to be employed in clustering problems since they are minimized by the mean (μ) as shown in [1]. Let us consider a distribution for the input \mathbf{x} so that this condition is fulfilled [19]:

$$\boldsymbol{\mu} = E[\mathbf{x}] = \arg \min_{\mathbf{y}} E[D_\phi(\mathbf{x}, \mathbf{y})] \quad (2)$$

The number of clusters will be noted N , and $\boldsymbol{\mu}_i$ will be the mean vector of the i -th cluster \mathcal{C}_i , $i \in \{1, \dots, N\}$. The cluster \mathcal{C}_i contains all points \mathbf{x} such that $\boldsymbol{\mu}_i$ has the minimum divergence with respect to \mathbf{x} :

$$\mathcal{C}_i = \left\{ \mathbf{x} \in \mathcal{S} \mid i = \arg \min_{j \in \{1, \dots, N\}} D_\phi(\mathbf{x}, \boldsymbol{\mu}_j) \right\} \quad (3)$$

Therefore (2) can be rewritten so as to split \mathcal{S} into N clusters \mathcal{C}_i :

$$\boldsymbol{\mu}_i = E[\mathbf{x} \mid \mathcal{C}_i] = \arg \min_{\mathbf{y}} E[D_\phi(\mathbf{x}, \mathbf{y}) \mid \mathcal{C}_i] \quad (4)$$

The GHBNG network can be regarded as a hierarchical tree structure, where each node is a Growing Neural Gas (GNG) network [5]. In addition to this, a growth control mechanism is added in order to limit the size of the GNGs. An initial growth phase is considered in the learning process, where units are inserted until the reduction of the quantization error is negligible. The subsequent convergence phase does not allow the insertion of new units. This means that each GNG contains a variable number of units, depending on the complexity of the data that it aims to represent. This is exemplified in Figure 1.

This kind of unsupervised neural network can be utilized in many applications. In this work, the GHBNG is applied to color quantization, where each neuron represents a color of the resulting codebook. Note that the size of the

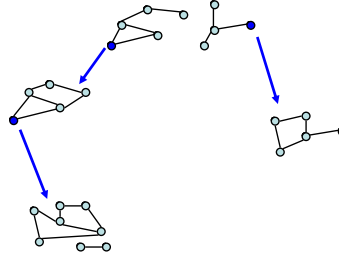


Fig. 1. GHBNG hierarchy, containing four graphs. The parent units are depicted in a darker tone.

codebook is automatically determined during the unsupervised learning process. Furthermore, the hierarchical architecture of the GHBNG provides different quantization rates, according to the number of levels of the neural hierarchy.

3 Experimental Results

Several experiments have been performed to investigate the capabilities of a color quantization system powered by a GHBNG. More precisely, the system has been trained and analyzed by using GHBNG models based on the five Bregman divergences mentioned in Section 2, namely: Squared Euclidean, Generalized I-Divergence, Itakura-Saito, Exponential Loss, and Logistic Loss. The images used in both the model training and testing processes consisted of four pictures frequently referenced in image processing research, namely baboon, house, lake, and Lena, which can be seen in Figures 2 and 3. The chosen research method is quasi-experimental, which is adequate for our exploratory purposes in this paper. Besides, as our GHBNG is a hierarchical tree-shaped model, a series of tests have been fulfilled using the Squared Euclidean Bregman divergence for one, two, three, and four-layer levels, in order to measure its performance improvement as the number of layer levels of neurons increases. All the networks have been trained using the same parameters tuned by using a trial and error-based empiric optimization method. These parameter values are shown in Table 2.

The training process starts by converting each image into a dataset that will be fed to the neural network, where each pixel of the image will represent a data sample including three features corresponding to the RGB components. Each dataset is composed of 262.144 samples as these images have a resolution of 512×512 pixels. The training of the models (one for each image and each Bregman divergence) consisted of performing 10 training rounds for each image and Bregman divergence so that the best model could be selected. The trees corresponding to these models had a maximum number of 4 levels and 50 neurons.

Table 2. Selected parameter values for the GHBNG model.

Parameter description	Values
Step size for the winning unit	$\epsilon_b = 0.2$
Step size for the neighbor unit	$\epsilon_n = 0.006$
Maximum edge for an edge	$a_{max} = 50$
New units insertion	$\lambda = 100$
Maximum number of neurons	$H_{max} = 50$
Reduction of the error variables	$\alpha = 0.5$
Error variable decay	$d = 0.995$

With the purpose of supporting a proper qualitative comparison of the color quantization results achieved by the GHBNG for each Bregman divergence, in Figure 2 are presented the differences images obtained from the subtraction of the original and the quantized images from the four original images we are considering. Since the quantization results were so close to the originals, the difference images presented here have been inverted and thresholded so they are visually noticeable. According to the figure, Itakura-Saito, Generalized I-Divergence, and Logistic Loss quantization errors seem to be slightly higher.

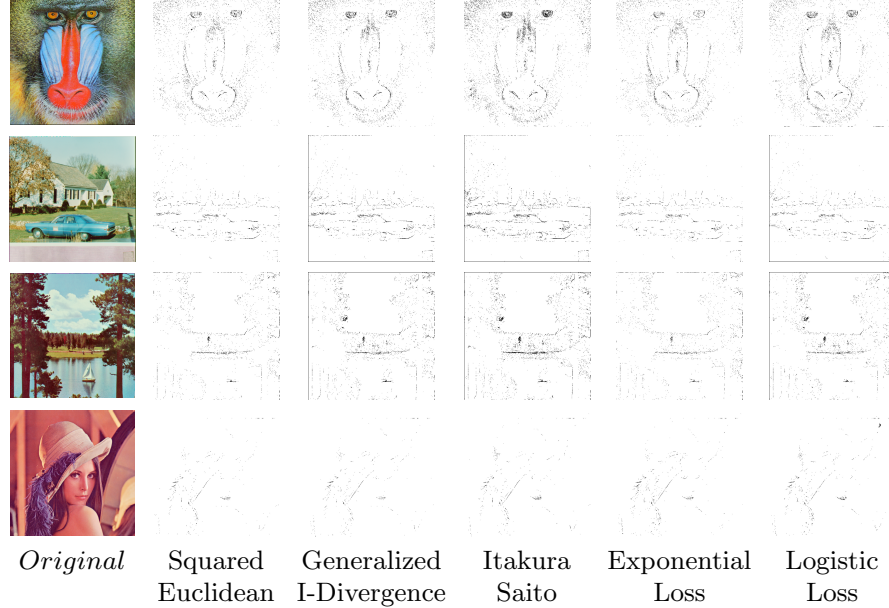


Fig. 2. Differences images for the 5 network models results corresponding to each of the 5 Bregman divergences. The darker the tone, the higher the quantization error.

When it comes to quantitative results, four widely used measures have been considered to test the performance of the color quantization task. The first one is the Mean Squared Error achieved by the different models after the training and test processes (Table 3).

Another popular performance measure is the Peak Signal-to-Noise Ratio (PSNR). PSNR values achieved by the models for each Bregman divergence can be checked in Table 3.

We have also used the structural similarity index measure (SSIM). Consequently, the SSIM values achieved by the models for each Bregman divergence are illustrated in Table 3.

The last performance measure considered in the experimental process is Normalized Cross-Correlation (NCC), and the values for this measure achieved by the models for each Bregman divergence are shown in Table 3.

Results presented in Table 3 indicate that the GHBNG model based in the Squared Euclidean Bregman divergence reaches the best MSE and PSNR values for most of the images. On the other hand, Table 3 suggests that despite the fact values being so close to each other, there is not a winning model in terms of the SSIM measure. NCC measure values again are close together. Nevertheless, this measure mostly suggests Squared Euclidean and Exponential Loss models as the most suitable for color quantization.

There are some measurements that can help us understand the behavior and performance of the GHBNG model. One of them is the training time (CPU time) for the model corresponding to each image and Bregman divergence. Another one is the compression rate (CR) reached by the GHBNG-generated images versus the original images. The results achieved by the GHBNG model for these measures are also illustrated in Table 3

As it has been stated before, the GHBNG network consists of a hierarchical tree-shaped model, so, for the sake of a better understanding of this model behavior, it is interesting to analyze its performance for different amounts of neuron layers which is equivalent to say as tree levels. In order to perform this task, Figure 3 illustrates how the performance of a GHBNG based in the Squared Euclidean Bregman divergence performance evolves as the number of layers or tree levels of the network varies from 1 to 4. From the images presented in Figure 3, it can be easily checked that the performance of the model in the color quantization task improves as the number of layers grows higher.

In order to clearly describe the features of the GHBNG model, a comparison between this model and other hierarchical self-organizing models has been performed. This comparison includes the training and test of the Growing Hierarchical Neural Gas (GHNG) [15] and the Growing Hierarchical Self-Organizing Map (GHSOM) [18] in the same conditions as the GHBNG. Thus, the parameters that GHNG and GHSOM have in common with GHBNG were adjusted to the same values as in GHBNG, whereas the other parameters were adjusted to values known to offer good performance in clustering tasks. Hence, the GHNG parameters were adjusted to the values present in Table 2 and the τ parameter was also set to $\tau = 0.1$, since the GHNG and GHBNG models share the same

Table 3. MSE, PSNR, SSIM, NCC, CPU time, and CR of the GHBNG model for each Bregman divergence. Best results are in bold. Standard deviations are in parenthesis.

		Baboon	House	Lake	Lena
MSE	SE	17.1121 (0.41)	5.7632 (0.23)	10.0209 (2.92)	11.8967 (0.52)
	GID	17.3387 (1.08)	6.2712 (0.20)	145.3983 (140.42)	57.4798 (8.51)
	IS	25.5235 (2.56)	8.0148 (0.47)	71.4242 (89.68)	59.9281 (3.92)
	EL	17.0709 (1.27)	5.9317 (0.30)	11.0030 (4.07)	14.2482 (6.33)
	LL	23.6376 (2.09)	6.1764 (0.25)	77.0336 (110.11)	53.8558 (3.11)
PSNR	SE	40.5701 (0.10)	45.2985 (0.17)	43.0130 (0.99)	42.1515 (0.19)
	GID	40.5194 (0.27)	44.9305 (0.14)	32.9426 (3.88)	35.3437 (0.57)
	IS	38.8523 (0.44)	43.8697 (0.25)	35.8290 (3.08)	35.1339 (0.29)
	EL	40.5895 (0.31)	45.1751 (0.22)	42.6649 (1.18)	41.6077 (1.36)
	LL	39.1797 (0.35)	44.9978 (0.17)	35.7026 (3.10)	35.5961 (0.25)
SSIM	SE	0.9910 (0.00)	0.9910 (0.00)	0.9919 (0.00)	0.9910 (0.00)
	GID	0.9905 (0.00)	0.9902 (0.00)	0.9931 (0.00)	0.9873 (0.00)
	IS	0.9858 (0.00)	0.9880 (0.00)	0.9850 (0.02)	0.9853 (0.00)
	EL	0.9913 (0.00)	0.9910 (0.00)	0.9920 (0.00)	0.9904 (0.00)
	LL	0.9909 (0.00)	0.9912 (0.00)	0.9935 (0.00)	0.9880 (0.00)
NCC	SE	0.9998 (0.00)	0.9999 (0.00)	1.0000 (0.00)	0.9999 (0.00)
	GID	0.9996 (0.00)	0.9999 (0.00)	1.0003 (0.00)	0.9995 (0.00)
	IS	0.9991 (0.00)	0.9998 (0.00)	0.9976 (0.01)	0.9993 (0.00)
	EL	0.9999 (0.00)	1.0000 (0.00)	1.0000 (0.00)	0.9993 (0.00)
	LL	0.9998 (0.00)	1.0000 (0.00)	0.9999 (0.00)	0.9996 (0.00)
CPU time	SE	96.9094 (1.27)	99.6016 (1.32)	97.8544 (4.11)	97.9143 (1.47)
	GID	114.9700 (7.01)	109.1663 (1.46)	115.2507 (3.72)	119.4984 (1.57)
	IS	107.0121 (5.83)	110.4900 (1.19)	111.2864 (14.49)	117.3473 (4.99)
	EL	113.0580 (3.04)	119.5402 (3.07)	115.8837 (1.52)	114.8192 (2.33)
	LL	130.2607 (6.60)	126.3012 (6.76)	130.8375 (2.68)	133.5190 (2.08)
CR	SE	1,71	1,71	1,85	1,71
	GID	1,71	1,71	1,71	1,71
	IS	1,71	1,71	1,71	1,71
	EL	1,71	1,71	1,85	1,71
	LL	1,71	1,71	1,71	1,71

Table 4. MSE, PSNR, SSIM, NCC, CPU time, and CR values obtained by the GHBNG, GHNG and GHSOM models. GHBNG results for the best performing Bregman divergence are shown. Best results are in bold. Standard deviations are in parenthesis.

		Baboon	House	Lake	Lena
MSE	GHBNG	17.0709 (1.27)	5.7632 (0.23)	10.0209 (2.92)	11.8967 (0.52)
	GHNG	16.6920 (0.41)	21.0434 (20.90)	12.3778 (0.72)	5.8046 (0.30)
	GHSOM	20.3121 (0.59)	7.6312 (0.36)	16.4853 (0.62)	8.8767 (0.71)
PSNR	GHBNG	40.5895 (0.31)	45.2985 (0.17)	43.0130 (0.99)	42.1515 (0.19)
	GHNG	40.6781 (0.11)	41.0350 (3.30)	41.9821 (0.25)	45.2694 (0.22)
	GHSOM	39.8261 (0.13)	44.0806 (0.21)	40.7340 (0.17)	43.4336 (0.38)
SSIM	GHBNG	0.9913 (0.00)	0.9912 (0.00)	0.9850 (0.02)	0.9910 (0.00)
	GHNG	0.9912 (0.00)	0.9777 (0.03)	0.9909 (0.00)	0.9910 (0.00)
	GHSOM	0.9891 (0.00)	0.9916 (0.00)	0.9879 (0.00)	0.9868 (0.00)
NCC	GHBNG	0.9999 (0.00)	1.0000 (0.00)	1.0000 (0.00)	0.9999 (0.00)
	GHNG	0.9998 (0.00)	1.0002 (0.00)	0.9999 (0.00)	0.9999 (0.00)
	GHSOM	0.9997 (0.00)	1.0000 (0.00)	0.9999 (0.00)	0.9999 (0.00)
CPU time	GHBNG	96.9094 (1.27)	99.6016 (1.32)	97.8544 (4.11)	97.9143 (1.47)
	GHNG	89.98 (3.97)	20.11 (2.34)	82.90 (2.88)	87.70 (1.64)
	GHSOM	209.68 (12.90)	57.22 (8.58)	208.80 (14.73)	193.10 (34.20)
CR	GHBNG	1.71	1.71	1.85	1.71
	GHNG	1.71	2.00	1.71	1.71
	GHSOM	1.71	1.85	1.85	1.71

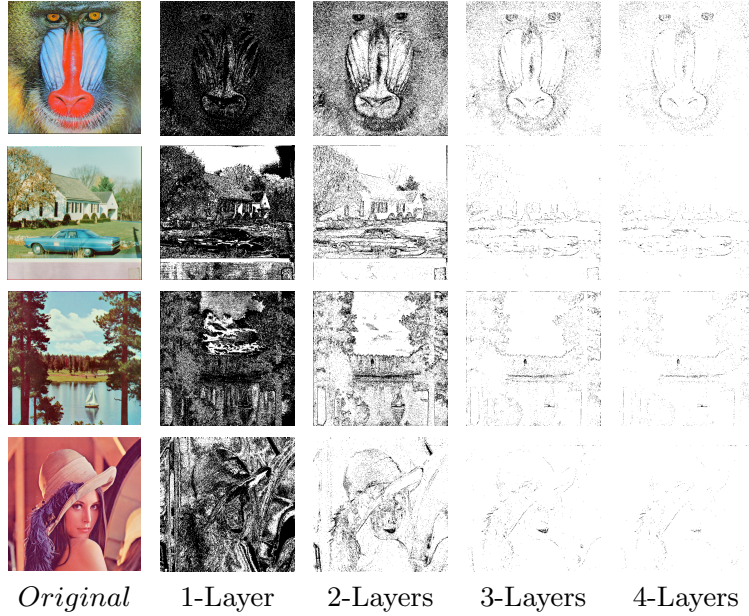


Fig. 3. Differences images for the 4 network models results and indicated number of layers for each model. The darker the tone, the higher the quantization error.

parameters. In the case of the GHSOM, the parameters τ_1 and τ_2 were adjusted to $\tau_1 = 0.001$ and $\tau_2 = 0.0001$ to obtain a similar number of neurons.

With the objective of offering a rigorous comparison of the GHBNG performance values against the selected competitors, the values corresponding to the best performing Bregman divergences for the GHBNG model are selected and compared with the GHNG and the GHSOM models in color quantization tasks considering the same popular performance measures used to test the GHBNG with Bregman divergences, namely MSE, PSNR SSIM, NCC, CPU time, and CR with the same test images: “Baboon”, “House”, “Lake”, and “Lena”. Results yielded by the tests appearing in Table 4, show that for the MSE performance measure GHNG performs better for “Baboon” and “Lena” images whilst GHBNG performs better for “House” and “Lake”. GHSOM values do not seem to stand out in any of the images considered. In the case of the PSNR, it occurs exactly the same as in the MSE performance measure. The GHBNG model presents the best values for the House” and “Lake” images, while the best values for the “Baboon” and “Lena” images have been obtained by the GHNG model. Once again, the GHSOM values do not highlight in any of the considered images. However, they are not as far from the values reached by the GHBNG and GHNG models as they were in the MSE performance measure. By carefully observing the values corresponding to the SSIM performance measure, we can check that the GHBNG behavior is better than the two models we have considered as its competitors except for the “Lake” image, in which the GHNG achieves better

results. The NCC values achieved by the GHBNG model attest that this model gets better results than the two competitor models considered in this experimentation process in all four of the test images. Finally, the GHNG obtains the best CPU time results, whereas the achieved CR is similar for the three models.

4 Conclusions

A novel approach for hierarchical color quantization based on the Growing Hierarchical Bregman Neural Gas (GHBNG) has been proposed in this work. This model is based on the Growing Hierarchical Neural Gas (GHNG) where Bregman divergences are taken into account. Thus, other distances different from the Euclidean distance can be considered and the best Bregman divergence for color quantization can be chosen. Furthermore, hierarchical color quantization is obtained after training the GHBNG with a color image, since the GHBNG creates a tree of graphs that represents the input data. Therefore, a different codebook can be found at each layer of the hierarchy so that the deeper the layer the larger the codebook obtained and the better the resulting color quantization, at the expense of a higher complexity of the codebook.

Experimental results obtained after training the GHBNG with well-known color images and different Bregman divergences show that aside from the squared Euclidean distance, the exponential loss represents a good choice for color quantization. In addition, the comparison performed with other hierarchical self-organizing models reveals a better performance of the GHBNG with Bregman divergences model over the GHNG and the GHSOM models in terms of performance measures, positioning the GHBNG with Bregman divergences as a powerful and avant-garde mathematical model for color quantization in digital images.

Acknowledgments. This work is partially supported by the Ministry of Science, Innovation, and Universities of Spain [grant RTI2018-094645-B-I00], project name Automated detection with low-cost hardware of unusual activities in video sequences, and by the Autonomous Government of Andalusia (Spain) under grant UMA18-FEDERJA-084, project name Detection of anomalous behavior agents by deep learning in low-cost video surveillance intelligent systems. All of them include funds from the European Regional Development Fund (ERDF). The authors thankfully the funding of the University of Málaga and acknowledge the computer resources and assistance provided by the SCBI (Supercomputing and Bioinformatics) center of the University of Málaga.

References

1. Banerjee, A., Merugu, S., Dhillon, I.S., Ghosh, J.: Clustering with Bregman divergences. *Journal of Machine Learning Research* 6, 1705–1749 (2005)
2. Bezdek, J.C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press (1981)

3. Bregman, L.: The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics* 7(3), 200–217 (1967)
4. Censor, Y., Zenios, S.: *Parallel Optimization: Theory, Algorithms, and Applications*. Oxford University Press (1998)
5. Fritzke, B.: A growing neural gas network learns topologies. *Advances in Neural Information Processing Systems* 7, 625–632 (1995)
6. Hertz, J., Krogh, A., Palmer, R.: *Introduction to the Theory of Neural Computation*. Addison-Wesley (1991)
7. Kohonen, T.: Self-organized formation of topologically correct feature maps. *Biological Cybernetics* 43(1), 59–69 (1982)
8. Kohonen, T.: The self-organizing map. *Proceedings of the IEEE* 78(9), 1464–1480 (1990)
9. Kohonen, T.: Essentials of the self-organizing map. *Neural Networks* 37, 52–65 (2013)
10. Linde, Y., Buzo, A., Gray, R.: An algorithm for vector quantizer design. *IEEE Transactions on Communications* 28(1), 84–95 (1980)
11. López-Rubio, E.: Probabilistic self-organizing maps for continuous data. *IEEE Transactions on Neural Networks* 21(10), 1543–1554 (2010)
12. López-Rubio, E.: Probabilistic self-organizing maps for qualitative data. *Neural Networks* 23(10), 1208–1225 (2010)
13. López-Rubio, E., Palomo-Ferrer, E.: Growing hierarchical probabilistic self-organizing graphs. *IEEE Transactions on Neural Networks* 22(7), 997–1008 (2011)
14. López-Rubio, E., Palomo-Ferrer, E.J., Ortiz-de Lazcano-Lobato, J.M., Vargas-González, M.C.: Dynamic topology learning with the probabilistic self-organizing graph. *Neurocomputing* 74(16), 2633–2648 (2011)
15. Palomo, E.J., López-Rubio, E.: The growing hierarchical neural gas self-organizing neural network. *IEEE Transactions on Neural Networks and Learning Systems* 28(9), 2000–2009 (2017)
16. Palomo, E.J., Molina-Cabello, M.A., López-Rubio, E., Luque-Baena, R.M.: A new self-organizing neural gas model based on Bregman divergences. *2018 International Joint Conference on Neural Networks (IJCNN)* pp. 1–8 (2018)
17. Palomo, E.J., Benito-Picazo, J., López-Rubio, E., Domínguez, E.: Unsupervised color quantization with the growing neural forest. In: Rojas, I., Joya, G., Catala, A. (eds.) *Advances in Computational Intelligence*. pp. 306–316. Springer International Publishing, Cham (2017)
18. Rauber, A., Merkl, D., Dittenbach, M.: The growing hierarchical self-organizing map: Exploratory analysis of high-dimensional data. *IEEE Transactions on Neural Networks* 13(6), 1331–1341 (2002)
19. Villmann, T., Haase, S.: Divergence-based vector quantization. *Neural Computation* 23, 1343–1392 (2011)